Forecasting and trading Bitcoin with machine learning techniques and a hybrid volatility/sentiment leverage

WEI, Mingzhe <http://orcid.org/0000-0002-8817-7788>, SERMPINIS, Georgios and STASINAKIS, Charalampos

Received: 13 December 2021 | Revised: 27 April 2022 | Accepted: 9 October 2022

DOI: 10.1002/for.2922

RESEARCH ARTICLE

WILEY

Forecasting and trading Bitcoin with machine learning techniques and a hybrid volatility/sentiment leverage

Mingzhe Wei | **Georgios Sermpinis** | **Charalampos Stasinakis**

Adam Smith Business School, University of Glasgow, Glasgow, UK

Correspondence
Mingzhe Wei, Sheffield Business School, Sheffield Hallam University, Sheffield, UK.
Email: wm9061@hallam.shu.ac.uk; W.Mingzhe@shu.ac.uk

Abstract

This paper explores the use of machine learning algorithms and narrative sentiments when applied to the task of forecasting and trading Bitcoin. The forecasting framework starts from the selection among 295 individual prediction models. Three machine learning approaches, namely, neural networks, support vector machines, and gradient boosting approach, are used to further improve the forecasting performance of individual models. By taking data-snooping bias into account, three different metrics are applied to examine the forecasting ability of each model. Our results suggest that the machine learning techniques always outperform the best individual model whereas the gradient boosting framework has the best performance among all the models. Finally, a time-varying leverage trading strategy combined with narrative sentiments and volatility is proposed to enhance trading performance. This suggests that the hybrid leverage strategy provides the highest Bitcoin profits consistently among all trading exercises.

KEYWORDS

cryptocurrencies, forecast combinations, narratives, trading strategies

1 | INTRODUCTION

Along with the explosive growth in machine learning (ML) algorithms and hardware development, prosperity in Fin-tech, Big-data, blockchain technology (BCH), and other high-tech fields is gradually changing the world. Carbonell et al. (1983) stated that the three primary research needs of ML methods are task-oriented, cognitive-simulated, and theoretical-analyzed. Based on the required tasks, ML methods can be categorized into classification problems, regression problems, anomaly-detection problems, clustering problems, and reinforce-learning problems (Alzubi et al., 2018). Compared with the classical ML algorithms, the modern-art neural networks (NNs), gradient boosting (GB), support vector regression (SVR), and other step-forward techniques enormously improve both computational efficiency and accuracy. The wide application of ML algorithms in stock, exchange-traded funds (ETF), and other conventional financial markets motivates our hypothetical success in the Bitcoin (BTC) market (Sermpinis et al., 2017).

Due to the highly volatile property of cryptocurrency and decentralization of BCH, prediction in BTC thereby

The data that support the findings of this study are available from the corresponding author upon reasonable request.

becomes the most challenging target. Prior studies focus on the traditional statistical models (e.g., GARCH), which are well documented (Gourieroux et al., 2020; Katsiampa, 2017; Zhang et al., 2021). However, only limited work has been done in cryptocurrency prediction using ML algorithms. In terms of NNs techniques, McNally et al. (2018) perform empirical results that the benchmark model is less accurate than NNs in BTC prediction. Ma et al. (2018) show GB styled methods are particularly efficient in P2P loan default prediction and the BTC market. Sun et al. (2020) display that SVR is more precise than benchmark models in forecasting BTC prices. Akyildirim et al. (2021) find ML classification algorithms have high accuracy in terms of cryptocurrency prediction. Chen et al. (2021) provide empirical evidence that a combination of ML techniques and economic and technology factors can predict the BTC exchange rate. A clear gap lies between past studies and recent exploration of ML applications in cryptocurrency, especially the BTC market.

Prior studies provide empirical evidence that the leverage trading strategy is profitable in the stock market (Sermpinis et al., 2014; Stasinakis et al., 2016). A more recent work by Kahraman and Tookes (2017) shows that leverage trading has a causal effect on market liquidity in the stock market. Inspired by these studies, the authors hence argue that a leverage trading strategy can be a solver on such occasions, allowing transactions when volatility is relatively low but avoiding trading when volatility is relatively high. Härdle et al. (2020) proposed that price dispersion driven by sentiment in the cryptocurrency market could be more significant than conventional financial markets. Unlike the traditional leverage strategy, the authors adopt the sentiment index as our leverage because an analysis of the influence of sentiment indicates that narratives and online media, like Twitter, Wikipedia, and Google Trends, are associated with BTC prices (Ciaian et al., 2016; Urquhart, 2018). The recent literature shows that BTC prices can be affected or even predicted by social media sentiment. Online media play a significant role in influencing behavior impacting the market and cannot be neglected (Mai et al., 2018; Zhang et al., 2018). With the growth of BCH and cryptocurrency, other media sources, like narratives or publications, should influence the BTC market.

Karalevicius et al. (2018) find that intraday BTC prices follow the direction of sentiment extracted from expertise news while leaving a short time gap for traders to react. Online board discussion is associated with extremely high volatility and jumps in BTC prices (Ahn & Kim, 2019). Caviggioli et al. (2020) argue that adopting BTC technology improves corporate reputation by studying Twitter data. Yao et al. (2019) find that news articles can influence BTC prices at a certain level. Azqueta-Gavaldón (2020) further find bidirectional causal relationships between narrative sentiment and BTC prices by applying a dynamic system model, whereas Süssmuth (2022) explains that mutual causality exists between web search dynamic and BTC prices before 2018. López-Cabarcos et al. (2021) provide a similar finding that the BTC market is connected with the investor sentiment and this relationship will be more significant in the stable period. These studies encourage the authors to employ a sentiment index constructed by narratives or formal publications as our leverages. The current study's objective is to explore the forecasting of BTC returns using a leverage trading strategy combined with the sentiment.

We have set up a two-step framework. At first, a large pool of conventional models is applied, including simple moving averages (SMA), exponential moving averages (EMA), autoregressions (AR), autoregressive moving averages (ARMA), and log prices to moving averages (PMA). Unlike traditional financial assets, conventional fundamental indicators cannot be found in cryptocurrencies. Thus, technical indicators could be one of the possible answers to the BTC prediction puzzle. Another reason should be attributed to the generalization and simplification of prediction. Our study examines whether the preliminary models have any predictive power in BTC prediction. We apply two dimensionality-reduction techniques and extract a certain number of critical factors for succeeding experiments, which are principal component analysis (PCA) and recursive feature elimination random forest (RFE-RF) algorithm. Finally, we use ML techniques, including multi-layer perceptron (MLP), a long-short term memory (LSTM), Extreme Gradient Boost Decision (XGB), Light Gradient Boost Decision (LBM), and SVR sets as forecast combination models to improve the predictive ability of individual models. The most accurate predictive model from the pool of individual models has been set as the benchmark based on three statistical measures, namely, the mean-squared error (MSE), root of mean-squared error (RMSE), and mean absolutely error (MAE). To formally examine the influence of over-fitting issue and data-snooping bias, we jointly apply three measures, which are the superior predictive ability (SPA) test of Hansen (2005), the modified Diebold and Mariano (MDM) (Harvey et al., 1997), and the model confidence set (MCS) of Hansen et al. (2011).

Secondly, we examine the usage of leverage trading strategy combined with sentiment and volatility and the traditional strategy to explore the profitability of forecast models. Unlike other financial markets, it is rather tricky for investors to understand and analyze fundamental

information without a certain level of knowledge. Thus, a trading strategy via technical analysis becomes a vitally important tool in the cryptocurrency market. Notably, we apply the time-varying leverage strategy in this study motivated by its robust performance in stock and exchange markets (see Sermpinis et al., 2014). Because of the subjectivity of sentiments, the polarity score is a popular proxy used in the natural language processing (NLP) field (Liu, 2012; Wei et al., 2020). Past studies have mainly used polarity or other sentiment scores to predict BTC prices (Ciaian et al., 2016; Guégan & Renault, 2021). Valencia et al. (2019) suggest that sentiment analysis using social media data can predict the direction of price movement for cryptocurrencies. López-Cabarcos et al. (2021) also prove that sentiment affects BTC volatility. Caferra (2022) also provided empirical evidence that investment strategies can be influenced by sentiment in the BTC market. By benchmarking the buy-and-hold strategy, we start with two different leverage trading strategies: pure volatility and pure sentiment leverage strategies. Moreover, we further apply a hybrid strategy by combing sentiment and volatility leverage. Our results show that all leverage strategies significantly improve trading performance and that the hybrid strategy outperforms other strategies.

The motivations behind our framework are the characteristics of BTC prices along with the unique flaws and merits of each model. Takaishi (2018) suggested that the distribution of daily BTC returns is multifractal with no volatility asymmetry. Like GARCH or ARIMA, traditional statistical models may not possess explanatory power on BTC prediction. Recent studies (e.g., Ji et al., 2019; Lahmiri & Bekiros, 2020; Mallqui & Fernandes, 2019; McNally et al., 2018) show that ML approaches are efficient and accurate in terms of cryptocurrency predictions. Schapire (2003) suggested that it is easier to obtain many rough rules of thumb than a highly accurate forecasting rule. Therefore, the forecast combination techniques lead to a more accurate result.

The results show that XGB is the best predictor among all the applied forecast combination models. Our investigation finds that all forecast combination models perform better than the benchmark in forecasting accuracy. By jointly applying three tests, our results control the data-snooping bias and over-fitting issue. We show that all forecasting combination models are more profitable than the benchmark model in terms of the overall trading performance. Among all models, we find that XGB has the best performance. The results are consistent with both the traditional trading strategy and the hybrid trading strategy. Our hybrid trading strategy generates much higher returns than the traditional trading strategy. Unlike previous papers, we consider the semantic definition of sentiment indices and extract reliable information sources from narratives.

The rest of the paper is organized as follows: Section 2 describes BTC returns and sentiment index. Section 3 summarizes the proposed individual models and combination forecast techniques. Forecasting and trading performance are given in Sections 4 and 5, respectively. Finally, Section 6 presents the conclusion of our study.

2 | DATA

2.1 | BTC

We have collected a total number of 1749 daily prices of BTC from January 1, 2014 to January 1, 2019 in three rolling forecasting exercises (F1, F2, and F3). The original data source can be found in Bitstamp. The data structure of this study is presented in Table 1.

We then obtain the daily series of returns in the following way:

$$R_t = \left(\frac{P_t}{P_{t-1}}\right) - 1. \quad (1)$$

Table 2 reflects the summary of descriptive statistics for BTC returns. The Jarque–Bera and augmented Dickey–Fuller (ADF) test also provide confirmative results and further justification for our statements. Meanwhile, the return series follows non-normal distribution and does not have a unit root at the 99% confidence level.

2.2 | Sentiment index

To construct the sentiment index, we have collected publications and news articles describing BTC from Factiva, containing massive reports, news, and other kinds of narratives from the worldwide business press, such as The Financial Times, The Economist, and things in that regard. Our main aim was to extract daily sentiment scores from these documents to generate a time-series sentiment index as the measure of leverage. In the current study, we collect a total number of 31,436 articles from January 1, 2014 to January 1, 2019. We first ran an ML algorithm for every article; this method is called Latent Dirichlet Allocation (LDA), developed by Blei et al. (2003). We have provided a brief introduction to LDA and the implementation process in Appendix S1. LDA is a widely used topic modelling technique (Chen

TABLE 1 Summary of dataset

Forecasting exercise	Data split	Number of observation	Start date	End date
F1	Total dataset	1232	01/01/2014	31/06/2017
	In-sample dataset	1132	01/01/2014	01/03/2017
	Out-of-sample dataset	110	02/03/2017	31/06/2017
F2	Total dataset	1395	01/07/2014	30/06/2018
	In-sample dataset	1255	01/07/2014	04/02/2018
	Out-of-sample dataset	140	05/02/2018	30/06/2018
F3	Total dataset	1389	01/01/2015	01/01/2019
	In-sample dataset	1250	01/01/2015	07/08/2018
	Out-of-sample dataset	139	08/08/2018	01/01/2019

Note: F2 is organized by rolling the dataset of F1 6 months ahead, and F3 is rolling forward 6 months ahead of F2. The different length in each period is caused by missing values or zero.

TABLE 2 Summary statistics of Bitcoin (BTC) returns

Return	Min	Mean	Max	SD	JB	ADF	S	K	LB (5)
BSP	−1	0.001	0.269	0.047	108***	0***	−5.51	122	0.33

Note: This table reports the sample statistics of cryptocurrency prices and returns. SD is the standard deviation; S is the skewness; K is the excess kurtosis; and ADF is the augmented Dickey–Fuller statistic. LB (5) are the Ljung–Box statistics with lag 5, respectively, distributed as χ^2 with n degrees of freedom, where n is the number of lags. JB is the Jarque–Bera test. The number of observations is 1749 for all series.
*Significance level: 10%.
**Significance level: 5%.
***Significance level: 1%.

et al., 2019; Feuerriegel & Pröllochs, 2018) with the distributions of word and topic, respectively, where documents are generated accordingly to these two distributions.

We obtain the sentiment of each article and further augment the sentiments of articles belonging to the top 10 topics on a daily basis. To generate time series of sentiments, we use a public library in the natural language process called TextBlob. TextBlob takes negation and modified words into account, measuring words with adjectives. To illustrate as an example, very good will be given a higher weight when calibrating the sentiment score of bad and not before good or bad will be assessed rightly as to their original meaning. Moreover, we can measure the sentiment from polarity (positive vs. negative, ranging from 1 to −1) and subjectivity (ranging from 0 to 1). Polarity scores reflect the sentimental attitude towards studied topics. When polarity scores are above zero, we believe the sentiment is positive and negative polarity scores vice versa.

To fit the NN models properly, we normalize the sentiment series by centralizing its mean to zero and unit variance, which is used to keep the magnitude of input data at the same level.

3 | FORECASTING MODELS

3.1 | Individual prediction models

As the first step, we apply a large number of single forecast models. As discussed in the earlier sections, quite a few cases focus on BTC prediction, and the majority of studies provide successful answers with high complexity models (Akyildirim et al., 2021; Ma et al., 2018; McNally et al., 2018). This study starts with a pool of linear models, including SMA, EMA, AR, ARMA, and PMA, in case of missing trials with easy models. Moreover, we take PMA ratios extended from the equilibrium model proposed by Detzel et al. (2021) into our model pool as the nonlinear component. PMA ratio is the difference between log prices and moving averages. In an economy like the cryptocurrency market, fundaments and other sources of information are difficult to find or trust. Under such circumstances, technical indicators constructed by past prices may become a unique weapon for investors. A detailed description of the models is provided in Table 3.

The total number of individual models is 295. In order to reduce the influence of the over-fitting issue caused by dimensionality issues, we apply the PCA

TABLE 3 Summary of individual forecast models

Linear models	Description	Total individual forecasts
SMA(q)	$E(R_t) = {}^{(R_t+\ldots+R_t)}/_{q}, q = 3\ldots30$	28
EMA(q′)	$E(R_t) = \frac{R_{t-1} + (1-\alpha')R_{t-2} + \ldots + (1-\alpha')^{q'-1}R_{t-q'}}{\alpha' + (1-\alpha') + \ldots + (1-\alpha')^{q'-1}}$, $q' = 3\ldots30$, $\alpha' = 2/(1+N_{days})$, N_{days} is the number of trading days	28
AR(q″)	$E(R_t) = \beta_0 + \sum_{i'=1}^{q''} \beta_{i'} R_{i-i'}, q'' = 1,\ldots,24, \beta_0, \beta_{i'}$ are the regression coefficients	24
ARMA (m, n)	$E(R_t) = \overline{\varphi}_0 + \sum_{j'=1}^{m'} \overline{\varphi}_{j'} R_{t-j'} + \overline{\alpha}_0 + \sum_{k'=1}^{n'} \overline{w}_{k'} \overline{\alpha}_{t-k'}$ $m', n' = 1\ldots15, \overline{\varphi}_0, \overline{\varphi}_{j'}$ are the regression coefficients, $\overline{\alpha}_0, \overline{\alpha}_{t-k'}$ are the residual terms, $\overline{w}_{k'}$ is the weights of the residual terms	210
PMA(L)	$PMA_t(L) = p_t - ma_t(L)$, where $ma_t(L) = \frac{1}{nL}\sum_{l=0}^{nL-1} p_{t-l}$	5

Note: The total number of individual inputs calculated is 290. In all the specifications above, R_t is the factor return at time t. P_t is the log price of the Bitcoin, and n *is* the number of days per week in $L = 1$, 2, 4, 10, and 20 weeks.
Abbreviations: AR, autoregressions; ARMA, autoregressive moving averages; EMA, exponential moving averages; PMA, log prices to moving averages; SMA, simple moving averages.

technique to extract the best set of predictors and discard high-correlated variables. PCA components account for 95% of the total variance, and only the selected components are used as inputs for all the remaining models. In total, we have 30 principal components selected from the linear pool of predictors. Previous studies, for instance, Chen et al. (2021) and Conn et al. (2019), show that the RF algorithm performs good feature selectivity. We have provided a general description of the RFE-RF process in Appendix S1. In order to make factor comparison with PCA more explanatory, we have also selected the best 30 factors based on individual feature importance. By applying RFE-RF, we have ranked predictors based on the order of their importance and set the best-performing benchmark model as our benchmark model. Table 4 summarizes the best predictor selected from the pool of individual models.

Table 4 shows that short-term lags may have better predictive power than longer lags. So then, we use ML algorithms to further improve the predictive ability of individual models.

TABLE 4 Summary of best individual predictor set

Forecasting exercise	MAE	MSE	RMSE
F1	EMA (3)	EMA (3)	EMA (3)
F2	SMA (1)	EMA (2)	PMA (2)
F3	PMA (1)	PMA (2)	PMA (1)

Note: The numbers in the parenthesis correspond to the lags in the individual model.
Abbreviations: EMA, exponential moving averages; MAE, mean absolutely error; MSE, mean-squared error; PMA, log prices to moving averages; RMSE, root of mean-squared error; SMA, simple moving averages.

3.2 | Combination forecast techniques

3.2.1 | MLP model

MLP is a traditional NN in the forecasting literature. Prior studies show the predictive power of MLP in BTC as well as conventional financial areas (Sin & Wang, 2017). The training process of MLP is relatively straightforward, that is, a perceptron with more than one set of layers. The input layer is generally considered the first step used to feed the training data into the model. The way from the input layer to the output layer is indirect, which shall go through an intermediary layer called the hidden layer. Finally, the output is the last step, producing the estimated value. For more details regarding the training process, the reader should refer to Shapiro (2000).

3.2.2 | LSTM

Similar to the recurrent NN (RNN), LSTM has a chain of repeating neural models with the above layers. However, to solve the long-term dependency issues, LSTM has added control gates, that is, the input gate, forget gate, and output gate. The main difference between RNN and LSTM is the cell state, which controls the information

regulated by three gates. Intuitively, sigmoid layers are used to determine how much information is to be restored. The range of [0,1] represents the kept information from nothing to all. Prior studies have shown the success of LSTM in many fields, but in this paper, we focus on their success in time-series prediction (Phaladisailoed & Numnonda, 2018). Compared with simple MLPs or other feed-forward NNs, LSTM has bidirectional neural connections. The latter implies that current data can be passed to the previous or the same layer. LSTM can thus keep the short-term memory as well as the long-term memory by control gates.

3.2.3 | SVR

SVR has been widely used in time-series prediction (Tay & Cao, 2001; Zhao et al., 2019). SVR follows the same principle as support vector machine (SVM) but aims to solve regression problems. Using kernel functions, SVR can project data into high-dimension space and find a hyperplane to control the error within a certain threshold. Unlike linear regression, the objective function of SVR is to minimize the L2-norm of the coefficient vector. In contrast to SVM, SVR tries to obtain as many samples as possible within decision boundary lines using slack variables. In a nutshell, SVR allows users to define their tolerance rate of errors and find an acceptable tolerance range by tuning.

3.2.4 | GBDT family: XGBoost (XGB) and LightGBM (LBM)

As one crucial branch of ensemble learning algorithms in the ML field, the gradient boost decision tree (GBDT) developed by Friedman (2001) is a multiple-task solver used in a myriad of aspects. According to the statistics of Kaggle, GBDT-based algorithms win the championship for more than half of ML competitions and are widely used in computer visualization, medicine, biology, and finance (Nobre & Neves, 2019; Rao et al., 2019; Wang & Gribskov, 2019). Intuitively, GBDT combines gradient boost (GB) and decision tree (DT). The former algorithm focuses on finding a strong learner $F(x)$ by aggregating a bunch of weak learners $T(x)$, whereas the latter is used to construct the judgement condition for learning power through iteration. Therefore, the training process of GBDT is additive. That is, the final prediction is based on the sum of previous predictions $(F(x) = F_1(x) + F_2(x) + ... + F_m(x))$.

Figure 1 provides a flowchart of GBDT structure, and the training process is described as follows (Friedman, 2001; Rao et al., 2019):

Input: Denote $\{x_i, y_i\}_{i=1}^{n}$ as training instances, where $x_i = (x_{1i}, x_{2i}, ..., x_{ki})$ denotes the features, k denotes the number of features, and y_i denotes the target value.

Step 1: Denote the initial constant value w and initialize the predictors as follows:

$F_0(x) = \arg\min_c \sum_{i=1}^{n} L(y_i, w)$, where $L(y_i, w)$ denotes the loss function.

Step 2: For data $i = 1,2,...,n$, we have the negative gradient or the residual along the gradient direction, $r_{im} = -\left[\frac{\partial L(y_i,\ F(x_i))}{\partial F(x_i)}\right]_{f(x)=f_{m-1}(x)}$ where m denotes the number of iterations.

Step 3: We fit sample instances into the initial tree $T(x_i; a_n)$ and obtain the parameter a_n through the least square method as $a_m \arg\min_{a,w} \sum_{i=1}^{n} (r_{im} - wT(x_i; a))^2$.

Step 4: To acquire the minimal loss function, the current weight of each base learner is described as

$$w_m = \arg\min_w \sum_{i=1}^{n} L(y_i, F_{m-1}(x) + wT(x_i; a_m)). \quad (2)$$

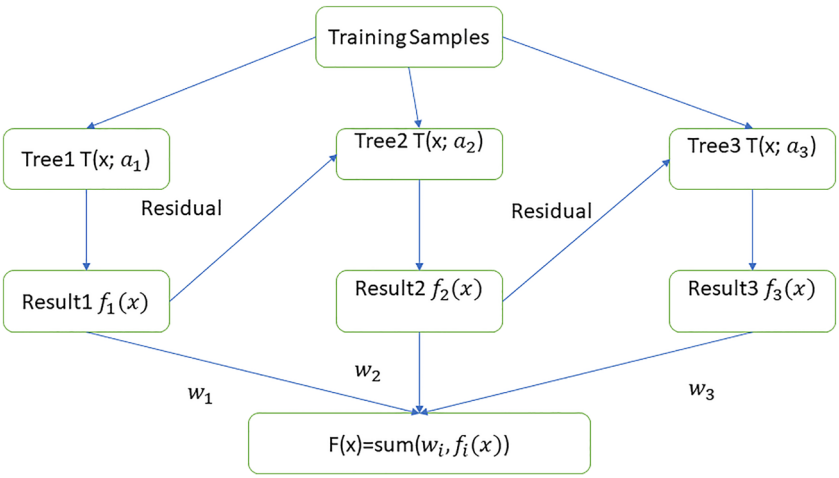

FIGURE 1 Flowchart of gradient boost decision tree (GBDT) structure *Note*: The residuals obtained from previous base learner is fed into the following base learner as training data (instance)

Step 5: The current prediction based on strong learner is given as follows

$$F_m(x) = F_{m-1}(x) + w_m T(x_i; a_n). \tag{3}$$

The above steps will keep running until the convergence condition or the specified iteration times are met.

XGBoost

Based on the structure of GBDT, Chen and Guestrin (2016) propose a scalable end-to-end gradient tree boosting model, XGB system, standing for "Extremely Gradient Boosting." Like other ML algorithms, XGB is designed to find a predictive model that best fits the training set (x_i) and target values (y_i). The following objective function needs to be minimized in order to measure how good the predictive model is:

$$OBJ(\theta) = L(\theta) + \Omega(\theta), \begin{Bmatrix} L(\theta) = \sum_{i=1}^{I} (y_i - \widehat{y}_i)^2, \\ \Omega(\theta) = \sum_{m=1}^{M} \Omega(f_m), \end{Bmatrix}, \tag{4}$$

where $L(\theta)$ denotes training loss function,[1] $\Omega(\theta)$ denotes regularization terms, $\widehat{y}_i$ is the prediction value, and m denotes the number of trees. The upper function ($L(\theta)$) is used to measure the forecasting ability of the tested model, and the below function ($\Omega(\theta)$) is used to control the model complexity. Particularly, the regularization term Ω is a function of the total number of leaves in the tree (N) and leaf weights (ω), which can be described as follows:

$$\Omega = \alpha \mathrm{N} + \frac{1}{2}\beta \|\omega\|^2, \tag{5}$$

where α denotes the complexity of leaves and β denotes the penalty parameter. The regularization term thus reduces the overfitting probability by leading to a predictive model with a simple structure. Intuitively, traditional optimization algorithms cannot be used for the objective function above.

For the tree ensemble model like XGB, the final prediction is the sum of the scores of each tree. Denote $\widehat{y}_i$ as the prediction in ith instance (x_i), and f_m denotes a tree structure to mainly improve our model in mth iteration. We therefore have the prediction score $\widehat{y}_i = \sum_{m=1}^{M} f_m(x_i)$ for x_i.

Then, second-order Taylor expanding is used to optimize the objective function as follows:

$$Obj = \sum_{j=1}^{N} \left[\left(G_j \omega_j + \frac{1}{2}\left(H_j + \beta \omega_j^2 \right) \right] + \alpha N, \right. \tag{6}$$

where I_j denotes the instance set for jth leaf, $G_j = \sum_{i \in I_j} g_i$ is a constant, denoting the sum of the first-order partial derivation of all samples in jth leaf, and $H_j = \sum_{i \in I_j} h_i$ is a constant, denoting the sum of the second-order partial derivation of all samples in jth leaf. Therefore, the optimization of the objective function is transferred into a minimum determination problem of a quadratic function.

Based on the definition of the loss function, XGB is capable of solving both classification and regression tasks. In general, XGB is an improved version of GBDT, optimizing the objective function by adding the regularization terms and increasing the prediction accuracy using the second-order Taylor expansion. Moreover, two more techniques are applied to tackle the overfitting issue during tree growth, which is shrinkage and column subsampling (see Chen and Guestrin, 2016, and Friedman et al., 2000, for a detailed discussion).

LightGBM

Similar to XGB, LBM is an open-source framework developed by Microsoft Research Asia in 2016 (Ke et al., 2017). Generally, LBM is designed to solve the lack of computation efficiency in mass data without losing much accuracy. Compared with XGB, LBM mainly has two advantages in time complexity reduction: finding the optimal splitting node and trees growth strategy. Regarding the first side, LBM has three aspects of improvement: reducing the number of splitting nodes, the size of training data, and the number of features. At first, LBM applies a histogram-based DT algorithm instead of a presorted approach in splitting points to reduce the number of splitting nodes. The principle of the histogram algorithm is discretizing the continuous floating-point eigenvalues into k number of small bins and constructing k-width histograms. The discrete values index the accumulation of histograms in each bin. Then, the sum of gradients and the number of samples in each bin, as required statistics, are gradually stored in the histogram. With the necessary statistics in histograms after the first traverse of data, it is possible to find the optimal segment point based on the discrete value indices. Compared with the presorted method, the histogram algorithm reduces the memory cost by storing only the discrete values. Productive discussion of both the presorted and

histogram-based methods is well documented already (Shafer et al., 1996; Ke et al., 2017). Secondly, LBM develops the gradient-based one side sampling (GOSS) technique to control the size of training instances. Unlike AdaBoost, no sample weights are given in GBDT, but the gradient of instances is also a good indicator for searching for the optimal split point. Intuitively, training instances with small gradients have relatively smaller training errors, indicating these parts of data are well-trained and should be abandoned. This is similar to GBDT in that large deviations from the target value will be penalized harder. In AdaBoost, the sample weight serves as a good indicator to determine the importance of samples. However, the data distribution may be distorted by the loss of instances and influence the accuracy of trained models. In order to keep the balance between reduction of data size and accuracy of learning decision trees, GOSS applies a constant multiplier to instances with low gradients when computing information gains. Exclusive Feature Bundling (EFB) is another critical technique in LBM, which is used to reduce the feature number. This method is inspired by the sparsity of high-dimension data and is designed to reduce the feature numbers by combining mutually exclusive features (values are simultaneously nonzero).

LBM applies a leaf-wise growth strategy with a depth controller, which searches for the maximum profit from leaf splitting, whereas a level-wise strategy splits every leaf. Unavoidably, the level-wise strategy used in XGB may generate redundant data and reduce computing efficiency. On the contrary, the leaf-wise growth strategy only focuses on the leaf with the most significant information gained on the same layer, enhancing algorithm speed. To control the possible overfitting issue, this method needs to manually set and tune the max depth of trees and minimum data in each leaf.

4 | STATISTICAL PERFORMANCE

In order to examine the statistical significance of each predictor, we employ three famous metrics, MSE, RMSE,

TABLE 5 Summary of out-of-sample statistical performance

Metrics	Forecasting exercise	Best	MLP	LSTM	ε-SVR	ν-SVR	XGB	LBM
	Panel A: Set of selected factors based on RFE-RF							
MAE	F1	0.0334	0.0185	0.0089	0.0080	0.0059	**0.0042**	0.0050
	F2	0.0416	0.0202	0.0083	0.0068	0.0057	**0.0050**	0.0347
	F3	0.0369	0.0093	0.0043	0.0087	0.0084	**0.0034**	0.0042
MSE	F1	0.00185	0.0011	0.00014	0.00011	0.00006	**0.00003**	0.00006
	F2	0.0025	0.0016	0.00037	0.00028	0.00027	**0.00006**	0.00092
	F3	0.00235	0.0013	0.00007	0.00017	0.00009	**0.00002**	0.00008
RMSE	F1	0.0430	0.0333	0.0119	0.0105	0.0079	**0.0056**	0.0075
	F2	0.0450	0.0403	0.0611	0.0167	0.0163	**0.0075**	0.0303
	F3	0.0484	0.0362	0.0087	0.0132	0.0095	**0.0042**	0.0089
	Panel B: Selected principal components							
MAE	F1	0.0334	0.0284	0.0173	0.0195	0.0179	**0.0056**	0.01925
	F2	0.0416	0.0304	0.0198	0.02129	0.02134	**0.01007**	0.02234
	F3	0.0369	0.0291	0.0256	0.03226	0.02809	**0.00702**	0.01735
MSE	F1	0.00185	0.00113	0.00050	0.00158	0.00064	**0.00005**	0.00028
	F2	0.0025	0.00162	0.00063	0.00089	0.00085	**0.00021**	0.00079
	F3	0.00235	0.00213	0.00062	0.00062	0.00072	**0.00035**	0.00072
RMSE	F1	0.0430	0.0334	0.02231	0.03978	0.02521	**0.00564**	0.0235
	F2	0.0450	0.0392	0.02506	0.02978	0.02914	**0.01457**	0.02804
	F3	0.0484	0.0462	0.02494	0.02495	0.02683	**0.01875**	0.02689

Note: Numbers in bold style denote the lowest statistics under corresponding metric.

Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MAE, mean absolutely error; MLP, multi-layer perceptron; MSE, mean-squared error; RMSE, root of mean-squared error; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.

and MAE[2] Instinctively, lower statistics indicate a more accurate prediction of the examined model. Table 5 reports the summary of out-of-sample statistical performance.

The above results show that the models' statistical ranking is consistent across three forecasting exercise periods and two sets of factors. Unsurprisingly, the best-performing benchmark selected from the pool of individual models is beaten by all ML models. In terms of the forecasting performance of ML algorithms, XGB provides the best statistical accuracy among the forecasting combination models. Although SVR sets are inferior to XGB in terms of out-of-sample performance, their predictive power is better than other models. The result is in line with several studies that suggest SVR can be a robust prediction tool supporting individual predictors (Sermpinis et al., 2014; Zhao et al., 2019). Finally, although LSTM falls short of the GBDT family, it has more accurate results than the benchmark, in line with recent experiments in BTC prediction (Ji et al., 2019; McNally et al., 2018). One possible reason could be our study's relatively short sample period.[3]

In order to formally validate the consistency of forecasting ability rank in the above results, we perform the MDM test suggested by Harvey et al. (1997). The MDM test statistic is calculated as follows:

$$MDM = T^{-\frac{1}{2}}\left[T+1-2k+T^{-1}k(k-1)\right]^{\frac{1}{2}DM}, \quad (7)$$

where T denotes the number of observations in the out-of-sample period and k denotes the number of step-ahead forecasts. Based on the forecasting performance of XGB, we apply MDM by benchmarking XGB and comparing it with the rest models one by one. A negative realization of the MDM test statistic implies XGB performs better than the second forecast model in prediction accuracy. The results of MDM tests are summarized in Table 6.

Unsurprisingly, the statistical outcomes of MDM from Table 7 confirm the consistency of the statistical

TABLE 6 Summary results of modified Diebold–Mariano statistics for MSE and MAE loss functions

Metrics	Best	MLP	LSTM	ε-SVR	ν-SVR	LBM	XGB
F1	Panel A: Set of selected factors based on RFE-RF						
MDM_1	−16.513***	−11.028***	−10.865***	−9.632***	−3.675***	−2.750**	-
MDM_2	−15.980***	−12.414***	−10.268***	−8.869***	−5.293***	−5.899***	-
F1	Panel B: Selected principal components						
MDM_1	−15.423***	−12.028***	−10.865***	−9.632***	−3.675***	−2.750**	-
MDM_2	−15.165***	−12.414***	−10.268***	−8.869***	−8.293***	−8.899***	-
F2	Panel A: Set of selected factors based on RFE-RF						
MDM_1	−17.416***	−11.325***	−8.816***	−4.858***	−3.330***	−10.709***	-
MDM_2	−19.413***	−11.144***	−7708***.	−4.329***	−3.223***	−9.291***	-
F2	Panel B: Selected principal components						
MDM_1	−20.471***	−11.437***	−13.620***	−14.725***	−14.808***	−13.651***	-
MDM_2	−17.527***	−11.267***	−11.830***	−11.011***	−10.987***	−9.845***	-
F3	Panel A: Set of selected factors based on RFE-RF						
MDM_1	−17.970***	−11.489***	−3.304***	−7.097***	−9.171***	−2.982**	-
MDM_2	−23.822***	−20.282***	−4.781***	−14.677***	−6.311***	−3.782***	-
F3	Panel B: Selected principal components						
MDM_1	−22.417***	−20.604***	−11.780***	−18.627***	−15.524***	−10.834***	-
MDM_2	−22.527***	−18.255***	−6.493***	−6.505***	−6.618***	−6.632***	-

Note: MDM_1 and MDM_2 are the statistics computed for MAE and MSE loss function, respectively. Missing sections represent the benchmark model.
Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MLP, multi-layer perceptron; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.
*Significance level: 10%.
**Significance level: 5%.
***Significance level: 1%.

TABLE 7 Summary results of MCS and SPA statistics

Metrics	Best	MLP	LSTM	ε-SVR	ν-SVR	LBM	XGB
F1	Panel A: Set of selected factors based on RFE-RF						
MCS	0	0	0.001	0.001	0.001	0.007	**1**
SPA	0	0	0	0.001	0.002	0.042	**0.522**
F1	Panel B: Selected principal components						
MCS	0	0	0	0	0	0	**0.564**
SPA	0	0	0	0	0	0	**0.758**
F2	Panel A: Set of selected factors based on RFE-RF						
MCS	0	0	0	0	0	0	**1**
SPA	0	0	0	0	0	0	**0.622**
F2	Panel B: Selected principal components						
MCS	0	0	0	0	0	0	**1**
SPA	0	0	0.001	0	0	0	**0.928**
F3	Panel A: Set of selected factors based on RFE-RF						
MCS	0	0	0	0	0	0	**1**
SPA	0	0	0	0	0	0	**0.868**
F3	Panel B: Selected principal components						
MCS	0	0	0	0	0	0	**1**
SPA	0	0	0.005	0.001	0.001	0.002	**0.613**

Note: MCS and SPA are the statistics computed for the model confidence set (MCS) of Hansen et al. (2011) and superior predictive ability test (SPA) of Hansen (2005), respectively. This table reports the *p*-value of aforementioned two statistics, high value of SPA indicates the benchmark model is superior to at least one of the other models and high value of MCS implies the benchmark model belongs to the set of top performing models. Numbers in bold style denote the top performing model set.
Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MLP, multi-layer perceptron; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.

ranking presented in Table 6. For both sets of factors (RFE-RF and PCA), we shed light on the predictive power of the GBDT family (XGB and LBM) because of all the negative statistics of the MDM test. Zhao et al. (2019) suggested that when the superiority of forecasting models suffers from data-snooping bias, the predictive performance may be attributed to luck. In order to further validate the superiority of the XGB model, we then apply two statistical tools, namely, the SPA test and the MCS test. The results are given in Table 7.

SPA test focuses on comparing the predictive abilities of multiple methods within a full set of models. High SPA *p*-values imply that at least one of the compared models may outperform the benchmark model. In our case, we examine the superior predictive power by benchmarking each model in turn and comparing it with the bundle of rest forecasting models. Based on the null hypothesis of SPA (no model is more accurate than the benchmark model), we declare that the predictive ability of XGB is superior to alternative models. All models from Table 8 are also used as benchmarks in turn in our second test (MCS), starting from the best-performing benchmark. MCS is a data-driven statistic that the more informative the data are, the fewer models are chosen (Hansen et al., 2011). By controlling the family-wise error, MCS determines the statistically insignificant set compared with the alternative model. High *p*-values indicate that the benchmark model should belong to the most accurate model set. The consistent results of both SPA and MCS across three forecasting exercises suggest the superior performance of XGB in terms of two sets of factors, which follows the logic of model forecasting performance. Moreover, the overall performance of predictive algorithms also suggests that encompassing robust forecasts can boost forecasting accuracy (Diebold & Pauly, 1990). In a nutshell, the outperformance of the XGB in the out-of-sample is genuine.

Through a comprehensive investigation, we provide evidence that ML techniques improve the predictive power of individual models. This is in line with our proposed hypothesis. The GBDT family has the best performance among the forecasting model pool.

TABLE 8 Summary results of out-of-sample traditional trading performance

Forecasting exercise	Metrics	Best	MLP	LSTM	ε-SVR	ν-SVR	LBM	XGB
Panel A: Set of selected factors based on RFE-RF								
F1	SR	0.0945	0.3665	0.3756	0.3873	0.4053	0.4189	0.5164
	AR	0.0111	0.021	0.0214	0.0221	0.0224	0.0228	0.0229
	SOR	0.1774	1.7485	1.9477	2.0374	2.099	2.1321	2.2788
	MDD	−0.2679	−0.3179	−0.3351	−0.349	−0.4063	−0.4087	−0.4505
	IR	0.1571	0.5795	0.5939	0.6123	0.6409	0.6624	0.8164
F2	SR	0.1031	0.2395	0.2489	0.297	0.3632	0.4047	0.481
	AR	0.0125	0.0219	0.0223	0.0241	0.031	0.0373	0.0378
	SOR	0.1998	1.7409	2.1774	2.4479	3.8398	4.9542	5.7908
	MDD	−0.4464	−0.526	−0.5454	−0.5493	−0.6349	−0.6938	−0.7633
	IR	0.163	0.3786	0.3935	0.4696	0.5742	0.6399	0.7605
F3	SR	0.0865	0.3472	0.3473	0.3567	0.3596	0.363	0.4664
	AR	0.0104	0.0216	0.0224	0.0228	0.0229	0.023	0.0231
	SOR	0.1741	1.8199	1.836	1.8549	1.8858	2.03	2.045
	MDD	−0.5377	−0.492	−0.5377	−0.5417	−0.5594	−0.5595	−0.5991
	IR	0.131	0.549	0.5492	0.564	0.5686	0.574	0.7374
Panel B: Selected principal components								
F1	SR	0.0945	0.3552	0.3691	0.3707	0.3821	0.4063	0.4945
	AR	0.0111	0.0183	0.0192	0.0196	0.02	0.0219	0.0238
	SOR	0.1774	1.2744	1.54	1.6322	1.708	2.0238	2.7958
	MDD	−0.2679	−0.4064	−0.4124	−0.4319	−0.4363	−0.4367	−0.5317
	IR	0.1571	0.5617	0.5836	0.5862	0.6041	0.6424	0.7819
F2	SR	0.1031	0.3808	0.3854	0.4098	0.4126	0.4143	0.4468
	AR	0.0125	0.021	0.0212	0.0223	0.0225	0.0226	0.0241
	SOR	0.1998	1.6755	1.6796	1.7388	1.8399	1.8557	2.1192
	MDD	−0.4898	−0.5454	−0.5631	−0.6009	−0.6305	−0.6321	−0.7307
	IR	0.163	0.6022	0.6093	0.6479	0.6524	0.6551	0.7064
F3	SR	0.0865	0.3208	0.3401	0.3461	0.3531	0.3648	0.4134
	AR	0.0104	0.0189	0.0194	0.0194	0.0196	0.0201	0.0226
	SOR	0.1741	1.189	1.2938	1.3033	1.3527	1.4823	2.0439
	MDD	−0.5377	−0.5383	−0.6155	−0.6247	−0.641	−0.6519	−0.6745
	IR	0.131	0.5072	0.5377	0.5472	0.5583	0.5767	0.6537

Note: SR stands for Sharp ratio, AR stands for annualized return, SOR stands for Sortino ratio, MDD stands for maximum drawdown, and IR stands for information ratio. Benchmark rates used in metrics are the annualized returns of buy-and-hold strategy in each forecasting exercise, which are 0.01826, 0.01185, and 0.01881.

Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MLP, multi-layer perceptron; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.

5 | TRADING PERFORMANCE

In order to examine the trading efficiency, we apply two approaches to our forecasting models. In Section 5.1, we apply the traditional trading strategy and a hybrid leverage trading strategy combining sentiment and volatility in Section 5.2. Formulas for both statistical and profitability measurements are given in Appendix S1.

5.1 | Trading performance of traditional trading strategy (L_T)

Intuitively, we choose to stay "*long*" when the forecast return at day t is above zero and stay "*short*" when the forecast return at day t is below zero. The "*long*" position is defined as buying BTC/USD at the current price, and the "*short*" position is defined as selling BTC/USD at the current price. Due to the lack of regulation in the cryptocurrency market, no unified trading cost is defined across different cryptocurrency exchanges. In particular, exchanges set variable standards of trading fees based on the payment area, payment type, and payment amount. For example, most cryptocurrency exchanges like Huobi

TABLE 9 Summary results of out-of-sample volatility (L_V) leveraged trading performance

Forecasting exercise	Metrics	Best	MLP	LSTM	ε-SVR	ν-SVR	LBM	XGB
Panel A: Set of selected factors based on RFE-RF								
F1	SR	0.2964	0.3837	0.394	0.3951	0.4019	0.4033	0.5443
	AR	0.0143	0.0262	0.0264	0.0278	0.028	0.028	0.0285
	SOR	0.3617	1.9878	2.0253	2.2054	2.2054	2.2926	2.3613
	MDD	−0.3936	−0.4427	−0.4692	−0.6898	−0.7684	−0.7843	−0.7877
	IR	0.4687	0.6067	0.623	0.6248	0.6355	0.6377	0.8607
F2	SR	0.1983	0.2419	0.2444	0.3386	0.3673	0.4455	0.6042
	AR	0.0163	0.0256	0.0268	0.0294	0.0388	0.0452	0.0462
	SOR	0.5821	2.144	2.8063	2.8396	4.3663	5.8737	7.1256
	MDD	−0.8703	−0.8818	−0.9243	−0.9322	−0.9539	−0.9936	−0.9942
	IR	0.3135	0.3825	0.3865	0.5355	0.5807	0.7044	0.9553
F3	SR	0.3441	0.398	0.3984	0.4154	0.4161	0.4162	0.5562
	AR	0.0146	0.0296	0.0318	0.0326	0.0327	0.0327	0.0328
	SOR	0.4782	2.6665	2.7737	2.9582	3.043	3.1201	3.1601
	MDD	−0.6954	−0.6999	−0.7337	−0.7644	−0.7776	−0.7819	−0.8875
	IR	0.5441	0.6293	0.6299	0.6569	0.6579	0.658	0.8794
Panel B: Selected principal components								
F1	SR	0.2964	0.4466	0.451	0.4581	0.4584	0.4591	0.4797
	AR	0.0143	0.0247	0.0259	0.0267	0.0269	0.0296	0.0303
	SOR	0.3617	1.9268	2.1219	2.3214	2.3913	2.8035	2.8818
	MDD	−0.4427	−0.5847	−0.5864	−0.5894	−0.5947	−0.6153	−0.7235
	IR	0.4687	0.7062	0.7132	0.7243	0.7247	0.7258	0.7585
F2	SR	0.3386	0.4484	0.4547	0.4553	0.4646	0.4658	0.4802
	AR	0.0163	0.0293	0.0303	0.0305	0.0306	0.0306	0.0328
	SOR	0.5821	2.2314	2.3158	2.3238	2.3613	2.4116	2.5739
	MDD	−0.8044	−0.8095	−0.8703	−0.9002	−0.9214	−0.9223	−0.9554
	IR	0.5355	0.709	0.7189	0.72	0.7346	0.7366	0.7592
F3	SR	0.3441	0.4414	0.4438	0.4595	0.4696	0.4713	0.4819
	AR	0.0146	0.0259	0.0272	0.0274	0.0277	0.0283	0.0315
	SOR	0.4782	2.2043	2.3159	2.5488	2.6624	2.814	3.1049
	MDD	−0.6999	−0.7389	−0.818	−0.8233	−0.8339	−0.877	−0.9125
	IR	0.5441	0.6978	0.7017	0.7265	0.7424	0.7452	0.762

Note: SR stands for Sharp ratio, AR stands for annualized return, SOR stands for Sortino ratio, MDD stands for maximum drawdown, and IR stands for information ratio. Benchmark rates used in metrics are the annualized returns of buy-and-hold strategy in each forecasting exercise, which are 0.01826, 0.01185, and 0.01881.

Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MLP, multi-layer perceptron; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.

or OKCoin used to charge no trading fees until the intense talk with Peoples Bank of China in 2017. Nonetheless, some exchanges preserve such rules and even set free costs for deposit and withdrawal fees, like SimpleFX and Coinfloor. Therefore, we do not consider the trading costs in our strategies. In Table 8, we present the out-of-sample trading performances of our models and NNs' techniques.

Forecasting models display positive trading performance for two sets of factors (see Table 8). Taking a look at the general ranking, the overall profitability performance of our models coincides with their forecasting performance. Forecasting combination techniques outperform the best-performing benchmark under all metrics in terms of model comparison. XGB is the superior model under most trading measures, which are annualized return (2.29%), Sharpe ratio (51.64%), Sortino ratio (2.2788), and information ratio (81.64%). Nonetheless, we can see maximum drawdown (MDD) of XGB (−45.05%) is also the highest because models with high returns come from high risk. We note that the MDD of all forecasting models ranges from 26% (the best-performing benchmark) to 45% (XGB), indicating that investors may lose nearly half of their funding for extreme cases. Compared with the performance of ML algorithms in the exchange market (−15%), the average MDD in the BTC market (−35%) is much higher (Sermpinis et al., 2014). Nonetheless, the average Sortino ratio is higher than 2 for all ML techniques, implying that investment in BTC is operating efficiently by taking those high risks. Across three forecasting exercises, F2 has the best performance whereas the worst subperiod is F1. The profits in BTC can be high, although it is also undeniable that investment in BTC should be cautious with its intensive volatility.

5.2 | Trading performance of volatility leverage, sentiment leverage, and hybrid leverage strategy

Because of the dramatic volatile property of BTC, we apply hybrid leverage based on two time-varying parameters, the first leverage based on daily volatility forecasts (L_V) and leverage based on sentiment (L_P). A detailed explanation of our strategy is given as follows.

The principle of the volatility forecasts (L_V) is to exploit transaction days when the return volatility is relatively low while reducing transaction days with extremely or relatively high volatility. In this way, we can quickly achieve the time-varying leverage by assigning inverse scale positions to recent risk measures while maintaining the information from market behavior.

At first, we employ a GJR (1,1) in the out-of-sample periods and forecast the 1 day ahead realized volatility of BTC returns. We further split the total test period into six subperiods, ranging from days with significantly low volatility to days with extremely high volatility. Based on the different volatility level of each day, we set up two parameters to classify our subperiods. The first parameter is the average (μ), which is the difference between the actual volatility in day t and the predicted for day $(t+1)$ and its corresponding standard deviation as the measure of volatility (σ). The parameters of our strategy are updated every 3 days by rolling forward the estimation period. That is, we classify periods when the difference is between μ and μ plus one σ as "Lower High Volatility." Similarly, we define periods with volatility larger than $(\mu+2\sigma)$ as "Extremely High Volatility" and periods with volatility between $(\mu+\sigma)$ and $\mu+2\sigma$ as "Medium High Volatility." Following the same method, we denote periods with volatility ranging from $(\mu-\sigma)$ to $\mu-2\sigma$ as "Medium Low Volatility" and periods with volatility below $\mu-2\sigma$ as "Extremely Low Volatility." As for the leverages (L_V) assigned for each period, we give 0 for periods with extremely high volatility and 2 for periods of extremely low volatility. Both parameters (μandσ) used in our method are updated every month by rolling forward the estimation period. This setup is consistent with the approach of Sermpinis et al. (2014).

Secondly, we construct sentiment leverage (L_P) based on the same approach of L_V. We replace μ and σ with the mean of polarity index (μ') and its standard deviation (σ'), respectively. Following the same classifying method of L_V, we split the test period into six subperiods and assign 0 for periods with extremely high volatility and 2 for periods of extremely low volatility. For robustness check, we then assign the leverages for each trading day based on the sign of the daily forecast. The goal is to boost profitability by exploiting more positive returns while shrinking losses incurred by negative returns.

1. If the forecast sign is positive (we are "long"), we apply leverage (L_V^+) of more than 1.
2. If the forecast sign is negative (we are "short"), we apply leverage (L_V^-) of less than 1.

5.2.1 | Volatility leverage (L_V)

L_V is available for each trading day. We apply L_V to each model and examine their trading performance following previous metrics. The results are given in Table 9.[4]

Table 9 summarizes trading performance for volatility leveraged strategy. Firstly, the trading performance of L_V is positive, and the ranking is consistent with its

performance in L_T. Across three forecasting exercises, F2 still takes the first place whereas F1 is the worst period. For model comparison, XGB still has the best performance under each measure, except for MDD (−78.77% in F1). For the RFE-RF factors, the overall risk grows higher because MDD of L_V ranges from −39% to −78% for F1. Similar results can be found in F2 and F3. Compared with the traditional trading strategy (Table 8), volatility leverage strategy amplifies high returns although it fails to shorten the corresponding risk. One possible reason could be attributed to the daily volatility variations. Although the volatility leverage strategy decreases the extreme negative returns, the variations from lowest returns to highest returns still grow much larger because of the significant increase in positive returns. In terms of ratio comparison between L_T and L_V, annualized returns increase above 0.2 times, whereas the Sortino ratio and Sharpe ratio at least increase above 0.04 times. In conclusion, the general performance of L_V is better than L_T.

5.2.2 | Sentiment leverage (L_P)

Sentiment has been widely used in financial areas, but recently, Chen and Hafner (2019) showed that the cryptocurrency market has a certain level of relationship with the news-driven sentiment. In this study, we apply a hybrid leverage strategy (L_H) combined with sentiment (L_P) and volatility to further improve our strategy's profitability. We also demonstrate sentiments (polarity and subjectivity) in Figure 2.

In order to generate L_H, we introduce two sentiment indices, polarity and subjectivity indices.[5] Polarity index is a prevalent indicator in sentiment analysis, commonly treated as a classifier for the trend moving by labelling either "positive" or "negative." However, recent articles do not notice the reliability of their information sources (Raju & Tarif, 2020). Nonetheless, it is vital to consider the subjectivity of comment or news when analyzing the sentiment (Subirats et al., 2018). In the present study, we employ two sentiment indices and use their literal definition as well as mathematical variations in our strategy. The first index can be interpreted as the different levels of attitude variation, whereas the second index is used to describe the subjectivity of collected documents. Naturally, we will only proceed to polarity measurement once the corresponding subjectivity score is above the threshold because measurement of the credibility of the information source into consideration is essential (Archak et al., 2011). Thus, the subjectivity index is regarded as the reliability measure of narratives, representing how much investors can trust (Nunkoo & Ramkissoon, 2012). Polarity gauges the sentiment from two sides, one for negative sentiment and the other for positive sentiment. Investors are thereby aware of the attitude variations from public recognition in cryptocurrency market and catch possible leverage opportunities. Nonetheless, it is possible to have unreliable sentiment sources. That is, narratives are full of too subjective descriptions or meaningless hypotheses. Under such circumstances, volatility leverage becomes the best option instead of sentiment leverage. Considering the range of subjectivity, we use

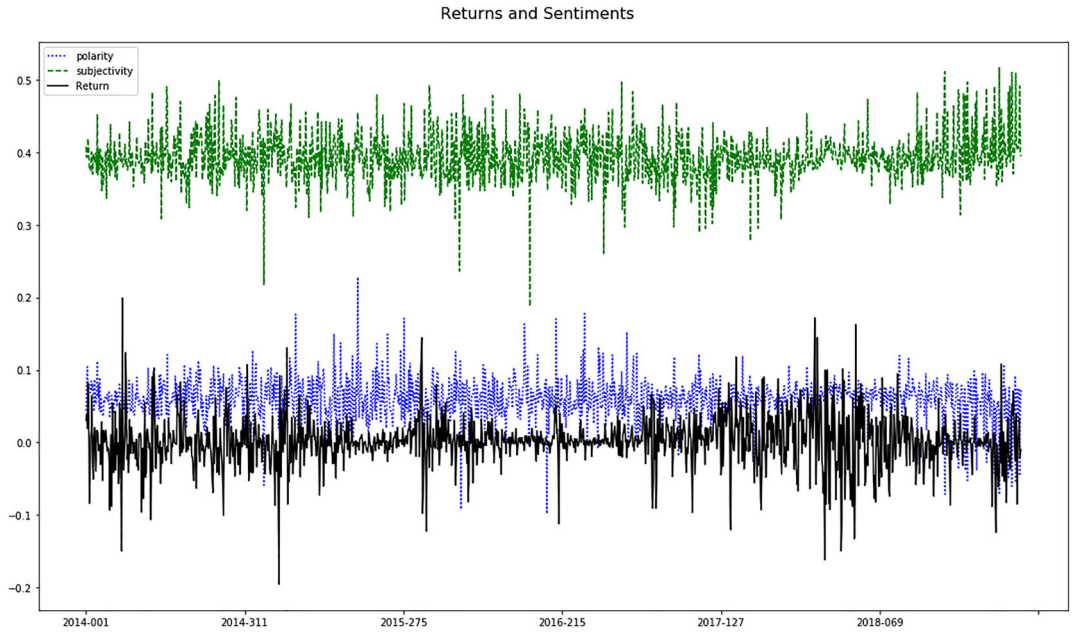

FIGURE 2 Sentiments and Bitcoin (BTC) returns *Note*: The x-axis denotes the number of days in a year

the moving average of 1 month as the threshold for subjectivity and employed volatility leverage when the subjectivity score is above the threshold.

The trading performance of L_P is summarized in Table 10.

As evidenced in Table 10, we note that the trading performance of L_P stays positive, and the general ranking is consistent with L_V and L_T. F2 has the best performance out of the three forecasting exercises, whereas F3 remains the worst period. As for model comparison, all ML models outperform the best-performing benchmark, and XGB is the superior model among all forecasting combination techniques. Similar to the performance of L_V, L_P improves the overall trading performance for most

TABLE 10 Summary results of out-of-sample sentiment (L_P) leveraged trading performance

Forecasting exercise	Metrics	Best	MLP	LSTM	ε-SVR	ν-SVR	LBM	XGB
Panel A: Set of selected factors based on RFE-RF								
F1	SR	0.4208	0.5321	0.5468	0.5473	0.5589	0.5949	0.7427
	AR	0.0157	0.0295	0.0298	0.0307	0.0312	0.0319	0.0323
	SOR	0.6425	3.4612	3.6569	3.7646	3.9059	3.9153	4.457
	MDD	−0.4321	−0.4769	−0.5184	−0.5186	−0.6058	−0.6096	−0.6608
	IR	0.6653	0.8414	0.8646	0.8654	0.8837	0.9406	1.1743
F2	SR	0.2765	0.3441	0.346	0.4978	0.502	0.5668	0.6974
	AR	0.0181	0.0308	0.0312	0.0342	0.0438	0.052	0.0525
	SOR	1.0624	3.4922	4.2986	5.0811	6.815	8.4348	9.9046
	MDD	−0.6511	−0.7275	−0.76	−0.789	−0.8197	−0.9525	−0.968
	IR	0.4371	0.544	0.5471	0.7871	0.7937	0.8962	1.1027
F3	SR	0.4331	0.4996	0.5122	0.516	0.5222	0.5242	0.7089
	AR	0.0145	0.0302	0.0309	0.0316	0.0316	0.0319	0.032
	SOR	0.5234	3.69	3.7381	3.8228	3.8382	4.2082	4.218
	MDD	−0.631	−0.7174	−0.7263	−0.7504	−0.7893	−0.8001	−0.8713
	IR	0.6848	0.7899	0.8099	0.8159	0.8257	0.8289	1.1209
Panel B: Selected principal components								
F1	SR	0.4208	0.606	0.6068	0.615	0.6168	0.6208	0.6837
	AR	0.0157	0.0261	0.0276	0.028	0.0286	0.0313	0.0339
	SOR	0.6425	2.8291	3.4197	3.4654	3.6127	4.1762	5.4981
	MDD	−0.4769	−0.5583	−0.574	−0.5924	−0.5972	−0.5994	−0.712
	IR	0.6653	0.9581	0.9594	0.9724	0.9753	0.9816	1.081
F2	SR	0.4978	0.5715	0.5775	0.5857	0.5864	0.5868	0.6175
	AR	0.0181	0.0295	0.0298	0.0313	0.0315	0.0316	0.034
	SOR	1.0624	3.451	3.458	3.4993	3.5783	3.5947	4.1888
	MDD	−0.6667	−0.7275	−0.7383	−0.8037	−0.8232	−0.8258	−0.8953
	IR	0.7871	0.9036	0.9131	0.9261	0.9271	0.9279	0.9764
F3	SR	0.4331	0.5505	0.5715	0.5935	0.5965	0.6003	0.6195
	AR	0.0145	0.0262	0.0269	0.0269	0.027	0.0278	0.0313
	SOR	0.5234	2.6373	2.8021	2.8386	2.9215	3.2667	4.2978
	MDD	−0.7263	−0.7493	−0.8571	−0.8603	−0.8833	−0.8914	−0.9155
	IR	0.6848	0.8704	0.9037	0.9384	0.9432	0.9491	0.9794

Note: SR stands for Sharp ratio, AR stands for annualized return, SOR stands for Sortino ratio, MDD stands for maximum drawdown, and IR stands for information ratio. Benchmark rates used in metrics are the annualized returns of buy-and-hold strategy in each forecasting exercise, which are 0.01826, 0.01185, and 0.01881.

Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MLP, multi-layer perceptron; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.

TABLE 11 Summary results of out-of-sample hybrid leveraged trading performance

Forecasting exercise	Metrics	Best	MLP	LSTM	ε-SVR	ν-SVR	LBM	XGB
Panel A: Set of selected factors based on RFE-RF								
F1	SR	0.428	0.526	0.5288	0.5399	0.5475	0.5493	0.7229
	AR	0.017	0.031	0.0329	0.0339	0.034	0.0343	0.0379
	SOR	0.820	3.573	3.9301	4.2511	4.2646	4.4093	4.737
	MDD	−0.484	−0.511	−0.5603	−0.5608	−0.7123	−0.727	−0.7287
	IR	0.677	0.831	0.8362	0.8537	0.8657	0.8685	1.143
F2	SR	0.248	0.313	0.3211	0.4491	0.464	0.5125	0.7115
	AR	0.019	0.033	0.0343	0.0376	0.0482	0.0569	0.0578
	SOR	1.172	4.163	5.5072	5.9065	7.9759	10.5002	12.6815
	MDD	−0.802	−0.863	−0.906	−0.9494	−0.9571	−0.9961	−0.9969
	IR	0.392	0.495	0.5077	0.71	0.7337	0.8104	1.125
F3	SR	0.443	0.471	0.4826	0.4922	0.4953	0.4976	0.6778
	AR	0.016	0.033	0.0351	0.0359	0.036	0.0361	0.0362
	SOR	0.781	4.491	4.503	4.5694	4.6788	4.992	5.0014
	MDD	−0.729	−0.784	−0.826	−0.852	−0.878	−0.887	−0.9414
	IR	0.701	0.7458	0.7631	0.7783	0.7832	0.7867	1.0717
Panel B: Selected principal components								
F1	SR	0.4286	0.5778	0.5783	0.5784	0.581	0.5814	0.6304
	AR	0.0171	0.0288	0.0303	0.0311	0.0316	0.0347	0.0373
	SOR	0.8201	3.1777	3.5642	3.8757	3.9319	4.8625	5.9532
	MDD	−0.5109	−0.6171	−0.6194	−0.6448	−0.6492	−0.6515	−0.7557
	IR	0.6776	0.9135	0.9143	0.9145	0.9186	0.9193	0.9967
F2	SR	0.4491	0.5536	0.5555	0.5556	0.5564	0.5666	0.5927
	AR	0.0193	0.0327	0.0332	0.0346	0.0348	0.035	0.0376
	SOR	1.172	3.8486	3.9254	4.0158	4.1704	4.2243	4.711
	MDD	−0.8579	−0.8632	−0.8823	−0.9154	−0.9351	−0.9354	−0.9618
	IR	0.71	0.8753	0.8783	0.8784	0.8798	0.8958	0.9371
F3	SR	0.4433	0.5321	0.5476	0.5638	0.5724	0.5759	0.5909
	AR	0.0162	0.0295	0.0303	0.0304	0.0304	0.0312	0.0353
	SOR	0.7814	3.2532	3.3726	3.4429	3.5415	3.9141	4.8992
	MDD	−0.7846	−0.8416	−0.9193	−0.921	−0.9258	−0.9326	−0.9645
	IR	0.7009	0.8414	0.8659	0.8915	0.905	0.9106	0.9343

Note: SR stands for Sharp ratio, AR stands for annualized return, SOR stands for Sortino ratio, MDD stands for maximum drawdown, and IR stands for information ratio. Benchmark rates used in metrics are the annualized returns of buy-and-hold strategy in each forecasting exercise, which are 0.01826, 0.01185, and 0.01881.

Abbreviations: LBM, Light Gradient Boost Decision; LSTM, long-short term memory; MLP, multi-layer perceptron; RFE-RF, recursive feature elimination random forest; SVR, support vector regression; XGB, Extreme Gradient Boost Decision.

profitability metrics. Taking F1 as example, annualized returns of XGB increases from 2.29% (L_T) to 3.23 (L_P), Sharpe ratio increases from 51.64% (L_T) to 74.27% (L_P), Sortino ratio increases from 2.2788 (L_T) to 4.457 (L_P), and information ratio increases from 81.64% (L_T) to 1.1743 (L_P). Although L_P still amplifies the general volatility, it seems to work better than L_V in solving extreme cases since MDD decreases from −78% (L_V) to −66% (L_P). In addition, XGB at least increases by 39% across three forecasting exercises under each profitability metric. In conclusion, we state the success of the sentiment leverage strategy.

5.2.3 | Hybrid leverage (L_H)

With both L_P and L_V, we describe the approach of hybrid strategy as follows:

$$L_H = 1_{S_{sub}} * L_H \begin{cases} L_H = L_P, \text{ when} S_{sub} \geq MA_{sub}(30) \\ L_H = L_V, \text{ otherwise} \end{cases}, \quad (8)$$

where $MA_{sub}(30)$ denotes the 30-day moving average of subjectivity scores, S_{sub} denotes the daily subjectivity, and L_P denotes leverage based on polarity. Once S_{sub} is lower than the threshold (a rejection of the usage of sentiment), we should depend on the volatility indicator. Similar to the volatility and sentiment leverage strategies, we then assign the leverages for each trading day based on the sign of the daily forecast.

We apply the hybrid trading strategy to each model and examine their trading performance by following previous metrics (see Table 11).

In conclusion, we argue that the hybrid trading strategy was successful based on the above findings. Compared with the traditional trading strategy, the annualized return of the hybrid strategy for each model is at least 1.4 times larger for both RFE-RF and PCA factors in all three forecasting exercises. For RFE-RF factors, XGB has the highest annualized return, Sharpe ratio, and information ratio, consistent with its performance in traditional strategy ranking for all three forecasting exercises. Similar results can be found in other ML techniques, which provides strong evidence that the application of sentiment leverage strategy significantly improves the profitability of forecasting techniques. Our findings align with the previous studies (Azqueta-Gavaldón, 2020; Karalevicius et al., 2018; Yao et al., 2019) that an interactive relationship exists between BTC and narratives, thus leading to the extraordinary profitability of the hybrid trading strategy.

6 | CONCLUSION

This study has examined the predictive power of forecast combination techniques and individual models. In terms of profitability examination, we propose a hybrid leverage trading strategy combining sentiment and volatility. Our investigation finds that forecast combination techniques outperform individual models in prediction accuracy. These results are roughly consistent with our hypothesis that ML techniques can improve the accuracy of simple individual models. Particularly, XGB has the best performance among all ML techniques. Moreover, our results are free of data-snooping bias through examining SPA, MCS, and MDM.

As for our examination of profitability, we apply two kinds of trading strategies, namely, a traditional strategy and three different leverage trading strategies: volatility leverage strategy, sentiment leverage strategy based on LDA, and hybrid leverage strategy by combining sentiment and volatility. Unsurprisingly, the annualized returns of ML techniques, especially for XGB, perform much better than other models for traditional strategy. Furthermore, with the application of a hybrid trading strategy, we find that the trading performance of all our forecasting models increases. These findings are in line with our hypothesis that strategies combined with sentiment indices can exaggerate the profitability of BTC.

In conclusion, XGB is the optimal forecast model based on remarkable trading performance and significant predictive accuracy. Furthermore, the success of our hybrid trading strategy indicates the importance of volatility and narrative sentiment in the cryptocurrency market. Although former research has suggested the usage of sentiment in the cryptocurrency market, the impact of online sentiment sources is limited (Urquhart, 2018). Prior studies showing sentiment as either a significant predictor or related factor apply their empirical results from 2010 to 2017 (Garcia & Schweitzer, 2015). Considering cryptocurrency's temporal influence and public recognition, exploring the sentiment indicator using a more recent period instead of a large scale or entirely early period data is essential. Due to limited information sources and uncertainty of new technology, preliminary indicators, such as Google Trends or post numbers on the website, may directly influence the early cryptocurrency market. We believe the development of BCH and consensus will strengthen the influence of the narrative sentiment index on the cryptocurrency sphere.

This study contributes to current literature in BTC forecasting and sheds light on trading strategies using sentiment and volatility leverage. Nonetheless, there is scope for further research on this topic. This paper adopts daily data rather than intraday data, which may not fully capture the price movements. Zhang et al. (2021) suggest that BTC returns can be used to predict BTC volatility by using aggregation of intraday information. In addition, Süssmuth (2022) provides evidence that Baidu–Google search statistics forecast BTC price dynamics at relatively high frequencies. Milunovich and Lee (2022) show that advanced ML techniques have high accuracy in predicting cryptocurrency exchange activity. Future work should focus on social media's influence on the aggregation of intraday trading information. Then, more hybrid

and advanced ML techniques should be explored in terms of their predictive performance and investment benefits. With the appropriate application of ML techniques, crypto-investment strategies could be more accurate and reliable and could be combined with mainstream trading approaches to capture the investment utilities and risk preferences of more investors.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Mingzhe Wei https://orcid.org/0000-0002-8817-7788

ENDNOTES

[1] This is a common MSE function. In addition, XGB is a highly compatible algorithm, allowing to employ a variety of metrics based on specific task.

[2] The statistical and trading performance metrics used in this study are standard in the literature. The relevant formulas are presented in Appendix S1.

[3] By grid search, we also try several parameters and several approaches of data preprocessing to seek for a beautiful prediction, but the result suggests LSTM cannot give a better result based on our sample. We do not deny the predictive ability of LSTM since its main usage is in NLP and recommendation algorithms where sufficient data are provided.

[4] To better understand our results, three tables illustrating the comparison between traditional strategy and each leverage strategy are provided in Appendix S1.

[5] We use the popular Python library, TextBlob, to generate sentiment indices. TextBlob is a popular and accurate tool in the NLP field.

REFERENCES

Ahn, Y., & Kim, D. (2019). Sentiment disagreement and bitcoin price fluctuations: A psycholinguistic approach. *Applied Economics Letters*, *27*(5), 412–416.

Akyildirim, E., Goncu, A., & Sensoy, A. (2021). Prediction of cryptocurrency returns using machine learning. *Annals of Operations Research*, *297*(1), 3–36. https://doi.org/10.1007/s10479-020-03575-y

Alzubi, J., Nayyar, A., & Kumar, A. (2018). Machine learning from theory to algorithms: An overview. *Journal of Physics: Conference Series*, *1142*, 012012. https://doi.org/10.1088/1742-6596/1142/1/012012

Archak, N., Ghose, A., & Ipeirotis, P. G. (2011). Deriving the pricing power of product features by mining consumer reviews. *Management Science*, *57*(8), 1485–1509. https://doi.org/10.1287/mnsc.1110.1370

Azqueta-Gavaldón, A. (2020). Causal inference between cryptocurrency narratives and prices: Evidence from a complex dynamic ecosystem. *Physica A: Statistical Mechanics and its Applications*, *537*, 122574. https://doi.org/10.1016/j.physa.2019.122574

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Caferra, R. (2022). Sentiment spillover and price dynamics: Information flow in the cryptocurrency and stock market. *Physica A: Statistical Mechanics and its Applications*, *593*, 126983. https://doi.org/10.1016/j.physa.2022.126983

Carbonell, J. G., Michalski, R. S., & Mitchell, T. M. (1983). An overview of machine learning. *Machine Learning*, *1*, 3–23.

Caviggioli, F., Lamberti, L., Landoni, P., & Meola, P. (2020). Technology adoption news and corporate reputation: Sentiment analysis about the introduction of Bitcoin. *Journal of Product & Brand Management*, *29*(7), 877–897. https://doi.org/10.1108/jpbm-03-2018-1774

Chen, C. Y. H., & Hafner, C. M. (2019). Sentiment-induced bubbles in the cryptocurrency market. *Journal of Risk and Financial Management*, *12*(2), 53. https://doi.org/10.3390/jrfm12020053

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794).

Chen, W., Xu, H., Jia, L., & Gao, Y. (2021). Machine learning model for Bitcoin exchange rate prediction using economic and technology determinants. *International Journal of Forecasting*, *37*(1), 28–43. https://doi.org/10.1016/j.ijforecast.2020.02.008

Chen, Y., Zhang, H., Liu, R., Ye, Z., & Lin, J. (2019). Experimental explorations on short text topic mining between LDA and NMF based schemes. *Knowledge-Based Systems*, *163*, 1–13. https://doi.org/10.1016/j.knosys.2018.08.011

Ciaian, P., Rajcaniova, M., & Kancs, D. A. (2016). The economics of BitCoin price formation. *Applied Economics*, *48*(19), 1799–1815. https://doi.org/10.1080/00036846.2015.1109038

Conn, D., Ngun, T., Li, G., & Ramirez, C. M. (2019). Fuzzy forests: Extending random forest feature selection for correlated, high-dimensional data. *Journal of Statistical Software*, *91*(1), 1–25. https://doi.org/10.18637/jss.v091.i09

Detzel, A., Liu, H., Strauss, J., Zhou, G., & Zhu, Y. (2021). Learning and predictability via technical analysis: Evidence from bitcoin and stocks with hard-to-value fundamentals. *Financial Management*, *50*(1), 107–137.

Diebold, F. X., & Pauly, P. (1990). The use of prior information in forecast combination. *International Journal of Forecasting*, *6*(4), 503–508.

Feuerriegel, S., & Pröllochs, N. (2018). Investor reaction to financial disclosures across topics: An application of latent Dirichlet allocation. *Decision Sciences*, *52*, 608–628. https://doi.org/10.1111/deci.12346

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting (with discussion and a rejoinder by the authors). *Annals of Statistics*, *28*(2), 337–407. https://doi.org/10.1214/aos/1016218223

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*, 1189–1232. https://doi.org/10.1214/aos/1013203451

Garcia, D., & Schweitzer, F. (2015). Social signals and algorithmic trading of Bitcoin. *Royal Society Open Science*, *2*(9), 150288. https://doi.org/10.1098/rsos.150288

Gourieroux, C., Hencic, A., & Jasiak, J. (2020). Forecast performance and bubble analysis in noncausal MAR(1, 1) processes. *Journal of Forecasting*, *40*(2), 301–326. https://doi.org/10.1002/for.2716

Guégan, D., & Renault, T. (2021). Does investor sentiment on social media provide robust information for Bitcoin returns predictability? *Finance Research Letters*, *38*, 101494.

Hansen, P. R. (2005). A test for superior predictive ability. *Journal of Business & Economic Statistics*, *23*(4), 365–380.

Hansen, P. R., Huang, Z., & Shek, H. H. (2011). Realized GARCH: A joint model for returns and realized measures of volatility. *Journal of Applied Econometrics*, *27*(6), 877–906.

Härdle, W., Harvey, C., & Reule, R. (2020). Understanding cryptocurrencies. *Journal of Financial Econometrics*, *18*(2), 181–208. https://doi.org/10.1093/jjfinec/nbz033

Harvey, D., Leybourne, S., & Newbold, P. (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting*, *13*(2), 281–291. https://doi.org/10.1016/S0169-2070(96)00719-4

Ji, S., Kim, J., & Im, H. (2019). A comparative study of Bitcoin price prediction using deep learning. *Mathematics*, *7*(10), 898. https://doi.org/10.3390/math7100898

Kahraman, B., & Tookes, H. E. (2017). Trader leverage and liquidity. *The Journal of Finance*, *72*(4), 1567–1610. https://doi.org/10.1111/jofi.12507

Karalevicius, V., Degrande, N., & De Weerdt, J. (2018). Using sentiment analysis to predict intraday Bitcoin price movements. *Journal of Risk Finance*, *19*(1), 56–75. https://doi.org/10.1108/JRF-06-2017-0092

Katsiampa, P. (2017). Volatility estimation for Bitcoin: A comparison of GARCH models. *Economics Letters*, *158*, 3–6. https://doi.org/10.1016/j.econlet.2017.06.023

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T. Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, *30*, 3146–3154.

Lahmiri, S., & Bekiros, S. (2020). Intelligent forecasting with machine learning trading systems in chaotic intraday Bitcoin market. *Chaos, Solitons & Fractals*, *133*, 109641. https://doi.org/10.1016/j.chaos.2020.109641

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, *5*(1), 1–167. https://doi.org/10.1007/978-3-031-02145-9

López-Cabarcos, M. Á., Pérez-Pico, A. M., Piñeiro-Chousa, J., & Šević, A. (2021). Bitcoin volatility, stock market and investor sentiment. Are they connected? *Finance Research Letters*, *38*, 101399. https://doi.org/10.1016/j.frl.2019.101399

Ma, X., Sha, J., Wang, D., Yu, Y., Yang, Q., & Niu, X. (2018). Study on a prediction of P2P network loan default based on the machine learning LightGBM and XGboost algorithms according to different high dimensional data cleaning. *Electronic Commerce Research and Applications*, *31*, 24–39. https://doi.org/10.1016/j.elerap.2018.08.002

Mai, F., Shan, Z., Bai, Q., Wang, X., & Chiang, R. H. (2018). How does social media impact Bitcoin value? A test of the silent majority hypothesis. *Journal of Management Information Systems*, *35*(1), 19–52. https://doi.org/10.1080/07421222.2018.1440774

Mallqui, D. C. A., & Fernandes, R. A. S. (2019). Predicting the direction, maximum, minimum and closing prices of daily Bitcoin exchange rate using machine learning techniques. *Applied Soft Computing*, *75*, 596–606. https://doi.org/10.1016/j.asoc.2018.11.038

McNally, S., Roche, J., & Caton, S. (2018). *Predicting the price of Bitcoin using machine learning* (p. 339). IEEE.

Milunovich, G., & Lee, S. A. (2022). Cryptocurrency exchanges: Predicting which markets will remain active. *Journal of Forecasting*, *41*(5), 945–955. https://doi.org/10.2139/ssrn.3799742

Nobre, J., & Neves, R. F. (2019). Combining principal component analysis, discrete wavelet transform and XGBoost to trade in the financial markets. *Expert Systems with Applications*, *125*, 181–194. https://doi.org/10.1016/j.eswa.2019.01.083

Nunkoo, R., & Ramkissoon, H. (2012). Power, trust, social exchange and community support. *Annals of Tourism Research*, *39*(2), 997–1023. https://doi.org/10.1016/j.annals.2011.11.017

Phaladisailoed, T., & Numnonda, T. (2018). Machine learning models comparison for Bitcoin price prediction. *2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE)*.

Raju, S. M., & Tarif, A. M. (2020). Real-time prediction of BITCOIN price using machine learning techniques and public sentiment analysis. *arXiv preprint arXiv:2006.14473*.

Rao, J., Liu, L., Tay, Y., Yang, W., Shi, P., & Lin, J. (2019). Bridging the gap between relevance matching and semantic matching for short text similarity modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 5373–5384).

Schapire, R. E. (2003). The boosting approach to machine learning: An overview. *Nonlinear Estimation and Classification*, *171*, 149–171. https://doi.org/10.1007/978-0-387-21579-2_9

Sermpinis, G., Stasinakis, C., & Dunis, C. (2014). Stochastic and genetic neural network combinations in trading and hybrid time-varying leverage effects. *Journal of International Financial Markets, Institutions and Money*, *30*, 21–54. https://doi.org/10.1016/j.intfin.2014.01.006

Sermpinis, G., Stasinakis, C., & Hassanniakalager, A. (2017). Reverse adaptive krill herd locally weighted support vector regression for forecasting and trading exchange traded funds. *European Journal of Operational Research*, *263*(2), 540–558. https://doi.org/10.1016/j.ejor.2017.06.019

Shafer, J., Agrawal, R., & Mehta, M. (1996, September). SPRINT: A scalable parallel classifier for data mining. In *Vldb* (Vol. 96, pp. 544–555).

Shapiro, A. F. (2000). A Hitchhiker's guide to the techniques of adaptive nonlinear models. *Insurance: Mathematics and Economics*, *26*(2–3), 119–132. https://doi.org/10.1016/S0167-6687(99)00058-X

Sin, E., & Wang, L. (2017, July). Bitcoin price prediction using ensembles of neural networks. In *2017 13th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)* (pp. 666–671). IEEE.

Stasinakis, C., Sermpinis, G., Psaradellis, I., & Verousis, T. (2016). Krill-herd support vector regression and heterogeneous autoregressive leverage: Evidence from forecasting and trading

commodities. *Quantitative Finance*, *16*(12), 1901–1915. https://doi.org/10.1080/14697688.2016.1211800

Subirats, L., Reguera, N., Bañón, A. M., Gómez-Zúñiga, B., Minguillón, J., & Armayones, M. (2018). Mining Facebook data of people with rare diseases: A content-based and temporal analysis. *International Journal of Environmental Research and Public Health*, *15*(9), 1877. https://doi.org/10.3390/ijerph15091877

Sun, X., Liu, M., & Sima, Z. (2020). A novel cryptocurrency price trend forecasting model based on LightGBM. *Finance Research Letters*, *32*, 101084. https://doi.org/10.1016/j.frl.2018.12.032

Süssmuth, B. (2022). The mutual predictability of Bitcoin and web search dynamics. *Journal of Forecasting*, *41*(3), 435–454.

Takaishi, T. (2018). Statistical properties and multifractality of bitcoin. *Physica A: Statistical Mechanics and its Applications*, *506*, 507–519. https://doi.org/10.1016/j.physa.2018.04.046

Tay, F. E., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Omega*, *29*(4), 309–317. https://doi.org/10.1016/S0305-0483(01)00026-3

Urquhart, A. (2018). What causes the attention of Bitcoin? *Economics Letters*, *166*, 40–44. https://doi.org/10.1016/j.econlet.2018.02.017

Valencia, F., Gómez-Espinosa, A., & Valdés-Aguirre, B. (2019). Price movement prediction of cryptocurrencies using sentiment analysis and machine learning. *Entropy*, *21*(6), 589. https://doi.org/10.3390/e21060589

Wang, J., & Gribskov, M. (2019). IRESpy: An XGBoost model for prediction of internal ribosome entry sites. *BMC Bioinformatics*, *20*(1), 1, 409–15. https://doi.org/10.1186/s12859-019-2999-7

Wei, J., Liao, J., Yang, Z., Wang, S., & Zhao, Q. (2020). BiLSTM with multi-polarity orthogonal attention for implicit sentiment analysis. *Neurocomputing*, *383*, 165–173. https://doi.org/10.1016/j.neucom.2019.11.054

Yao, W., Xu, K., & Li, Q. (2019). *Exploring the influence of news articles on Bitcoin price with machine learning* (p. 1). IEEE.

Zhang, W., Wang, P., Li, X., & Shen, D. (2018). Quantifying the cross-correlations between online searches and Bitcoin market. *Physica A: Statistical Mechanics and its Applications*, *509*, 657–672. https://doi.org/10.1016/j.physa.2018.06.073

Zhang, Y., He, M., Wen, D., & Wang, Y. (2021. (Forthcoming). Forecasting Bitcoin volatility: A new insight from the threshold regression model. *Journal of Forecasting*, *41*, 633–652. https://doi.org/10.1002/for.2822

Zhao, Y., Stasinakis, C., Sermpinis, G., & Fernandes, F. D. S. (2019). Revisiting Fama–French factors' predictability with Bayesian modelling and copula-based portfolio optimization. *International Journal of Finance & Economics*, *24*(4), 1443–1463. https://doi.org/10.1002/ijfe.1742

AUTHOR BIOGRAPHIES

Mingzhe Wei is a lecturer in fintech in the Department of Finance, Accounting and Business Systems, Sheffield Business School, Sheffield Hallam University. He holds a BSc in Mathematics with Finance from the University of Liverpool and Xi'an Jiaotong Liverpool University, an MSc in Finance from the University of Leicester, and a PhD in Accounting and Finance from Adam Smith Business School, University of Glasgow. His research interests lie in financial forecasting, machine learning and artificial intelligence, and financial technology (fintech).

Georgios Sermpinis is a professor of finance in Adam Smith Business School, University of Glasgow. He holds degrees from the National and Kapodistrian University of Athens and the Liverpool John Moores University. He previously worked at the University of Bedfordshire and Liverpool John Moores University. During his career, he has offered consultancies and provided seminars for major banks such as Goldman Sachs, BNP Paribas, Santander, and Societe Generale. His research interests lie in the areas of financial forecasting and trading, machine learning, econometrics, and operations research.

Charalampos Stasinakis is a professor of finance in Adam Smith Business School, University of Glasgow. He holds a BSc and an MSc in Computer and Electrical Engineering from the National Technical University of Athens and a PhD in Quantitative Finance from Adam Smith Business School, Economics, University of Glasgow. He previously worked in Bournemouth University while he has delivered seminars/visiting lectures in institutions, such as Heriot-Watt and Strathclyde University. His research interests lie in the areas of financial forecasting, machine learning and artificial intelligence, operations research, and financial technology.

SUPPORTING INFORMATION

Additional supporting information can be found online in the Supporting Information section at the end of this article.

How to cite this article: Wei, M., Sermpinis, G., & Stasinakis, C. (2022). Forecasting and trading Bitcoin with machine learning techniques and a hybrid volatility/sentiment leverage. *Journal of Forecasting*, 1–20. https://doi.org/10.1002/for.2922